PRBIT - Princípy bezpečnosti informačných technológií

Report - Domáca úloha č.4

Autor: Marek Čederle

Cvičenie: Pondelok 17:00

Použité príkazy a ich vysvetlenie

Zadanie č.1

- Úloha 1: nastavenia siete
  - Zistiť IP adresu a MAC adresu
  - Zistiť spôsob získania IP adresy
  - Zistiť smerovaciu tabuľku
  - Zistiť nastavenia DNS
  - Preveriť dostupnosť siete a Internetu pomocou ICMP.

IP adresu, MAC adresu a informáciu o tom, či používame DHCP zistíme pomocou nasledujúceho príkazu:

```
ip addr
```

Identifikujeme rozhranie, ktoré nás zaujíma, v tomto prípade `enp0s3` resp. `eth0`. Zistíme IP adresu, MAC adresu a informáciu o DHCP.

- `inet` - IP adresa aj s prefixom (maskou siete)
  - `dynamic` - IP adresa je pridelená pomocou DHCP
- `link/ether` - MAC adresa

```
[rocky@rocky-student-6 ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether fa:f1:17:82:04:c8 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 10.103.1.17/16 brd 10.103.255.255 scope global dynamic noprefixroute eth0
       valid_lft 49343sec preferred_lft 49343sec
    inet6 fe80::f8f1:17ff:fe82:4c8/64 scope link
       valid_lft forever preferred_lft forever
[rocky@rocky-student-6 ~]$ _
```

Na zistenie smerovacej tabuľky môžeme použiť jeden z nasledujúcich príkazov:

```
route -n
# alebo
ip route
```

```
[rocky@rocky-student-6 ~]$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.103.0.1      0.0.0.0         UG    100    0        0 eth0
10.103.0.0      0.0.0.0         255.255.0.0     U     100    0        0 eth0
10.105.0.0      0.0.0.0         255.255.0.0     U     100    0        0 eth0
169.254.169.254 10.103.1.3      255.255.255.255 UGH   100    0        0 eth0
[rocky@rocky-student-6 ~]$ ip route
default via 10.103.0.1 dev eth0 proto dhcp src 10.103.1.17 metric 100
10.103.0.0/16 dev eth0 proto kernel scope link src 10.103.1.17 metric 100
10.105.0.0/16 dev eth0 proto dhcp scope link src 10.103.1.17 metric 100
169.254.169.254 via 10.103.1.3 dev eth0 proto dhcp src 10.103.1.17 metric 100
[rocky@rocky-student-6 ~]$ _
```

Na zistenie DNS serverov si vieme vypísať konfiguračný súbor `/etc/resolv.conf`:

```
cat /etc/resolv.conf
```

Dostupnosť sieťových služieb môžeme zistiť pomocou príkazu `ping`, ktorý používa protokol `ICMP`:

```
ping -c 4 1.1.1.1
```

- `-c 4` - počet odoslaných paketov (count 4)
- `1.1.1.1` - IP adresa, ktorú chceme pingnúť (v tomto prípade verejný Cloudflare DNS server)

Zadanie č.2

- Úloha 2: nastavenia firewall-u:
  - Povoliť prichádzajúcu komunikáciu, ktorá patrí do existujúcich vytvorených spojení a komunikáciu súvisiacu (related) s odchádzajúcimi požiadavkami.
  - Povoliť prichádzajúcu ICMP komunikáciu.
  - Povoliť komunikáciu z lokálneho (loopback) rozhrania.
  - Povoliť prichádzajúce **požiadavky** na nové spojenia na službu SSH.
  - Povoliť prichádzajúce **požiadavky** na nové spojenia pre IP adresy z lokálnej siete (kde je pripojená VM) na všetky známe (well-known) porty.
  - Ostatnú prichádzajúcu komunikáciu logovať a zakázať.
  - Nastaviť aplikovanie pravidiel firewall-u pri štarte OS.

Všetky pravidlá vieme nastaviť pomocou príkazu `iptables`.

```
# Povolenie existujúcich spojení a súvisiacej komunikácie
sudo iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT

# Povolenie prichádzajúcej ICMP komunikácie
sudo iptables -A INPUT -i eth0 -p icmp -j ACCEPT

# Povolenie loopback rozhrania (všade)
sudo iptables -A INPUT -i lo -j ACCEPT
sudo iptables -A OUTPUT -o lo -j ACCEPT
sudo iptables -A FORWARD -i lo -j ACCEPT
sudo iptables -A FORWARD -o lo -j ACCEPT

# Povolenie nových SSH spojení
sudo iptables -A INPUT -i eth0 -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT

# Povolenie novej komunikácie z lokálnej siete na známe porty
sudo iptables -A INPUT -i eth0 -p tcp -m state --state NEW -s 10.103.0.0/16 --dport 0:1023 -j ACCEPT
sudo iptables -A INPUT -i eth0 -p udp -m state --state NEW -s 10.103.0.0/16 --dport 0:1023 -j ACCEPT

# Logovanie všetkých prichádzajúcich paketov
sudo iptables -A INPUT -j LOG

# Zakázanie všetkého ostatného, čo nebolo definované vyššie
sudo iptables -P INPUT DROP

# Uloženie nastavení aby sa načítali a aplikovali pravidlá aj po reštarte
service iptables save
```

```
[rocky@rocky-student-6 ~]$ sudo iptables -vnL
Chain INPUT (policy DROP 89 packets, 3440 bytes)
 pkts bytes target     prot opt in     out     source               destination
 5673  512K ACCEPT     all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
   28  2352 ACCEPT     icmp --  eth0   *       0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
   11   572 ACCEPT     tcp  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
    0     0 ACCEPT     tcp  --  eth0   *       10.103.0.0/16        0.0.0.0/0           state NEW tcp dpts:0:1023
    0     0 ACCEPT     udp  --  eth0   *       10.103.0.0/16        0.0.0.0/0           state NEW udp dpts:0:1023
  156  5584 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0           LOG flags 0 level 4

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT 2594 packets, 348K bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0
[rocky@rocky-student-6 ~]$
```

Zadanie č.3

- Úloha 3: nastavenia SSH
  - Nastaviť službu SSH na Vašom virtuálnom stroji, aby:
  - neumožnila prihlásenie používateľa 'root',
  - neumožnila autentifikáciu heslom (iba kľúčom),
  - neumožnila presmerovanie X11.

Všetko vyriešime editovaním konfiguračného súboru pre SSH deamona.

```
sudo nano /etc/ssh/sshd_config
```

Treba zmeniť nasledujúce hodnoty:

Pred	Po
PermitRootLogin yes	PermitRootLogin no
PasswordAuthentication yes	PasswordAuthentication no
X11Forwarding yes	X11Forwarding no

Následne je treba reštartovať SSH deamona, aby sa zmeny prejavili:

```
sudo systemctl restart sshd
```

```
PS C:\Users\marek> ssh root@147.175.150.103 -p 8028 -i "C:\Users\marek\Documents\prbit_vm_access_key\student6_id_rsa"
root@147.175.150.103: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
PS C:\Users\marek>
```

Zadanie č.4

- Úloha 4: SSH autentifikácia kľúčom
  - Nastaviť SSH na stroji 'student' tak, aby umožnila autentifikáciu bez zadávania hesla (pomocou kľúča).

Miesto toho aby som použil `student` server, tak som si spravil inštanciu na Linode cloude, kde som si zriadil Ubuntu 24.04 LTS server. A tam som si nastavil autentifikáciu pomocou SSH kľúčov podobne ako v úlohe 3 s tým, že som si vygeneroval kľúč na svojom počítači a následne som ho nahral na server pomocou ich webového rozhrania. Na generovanie a nahratie kľúčov som z časti použil ich oficiálny návod. Príkazy použité po pripojený na server ako root (verejný kľúč už je na servery nahraný):

```
# Základný setup
sudo apt update && sudo apt upgrade -y
sudo apt update && sudo apt install neofetch net-tools -y
sudo reboot

# Vytvorenie testovacieho používateľa
sudo useradd -m -s /bin/bash --group sudo prbit
sudo usermod -aG sudo prbit
passwd prbit

# Skopírovanie verejného kľúča do adresára používateľa (prihlásený ako root )
mkdir /home/prbit/.ssh
touch /home/prbit/.ssh/authorized_keys
cat ~/.ssh/authorized_keys > /home/prbit/.ssh/authorized_keys

# Editovanie konfiguračného súboru SSH ako v úlohe 3
sudo nano /etc/ssh/sshd_config
```

Pred	Po
PermitRootLogin yes	PermitRootLogin no
PasswordAuthentication yes	PasswordAuthentication no
X11Forwarding yes	X11Forwarding no

```
# Reštartovanie SSH deamona
sudo systemctl restart ssh
```

```
PS C:\Users\marek> ssh prbit@[redacted] -i C:\Users\marek\.ssh\id_ed25519
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Mon Oct  7 09:25:03 PM UTC 2024

  System load:           0.0
  Usage of /:            12.1% of 24.04GB
  Memory usage:          16%
  Swap usage:            0%
  Processes:             106
  Users logged in:       0
  IPv4 address for eth0: [redacted]
  IPv6 address for eth0: [redacted]


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Mon Oct  7 21:25:04 2024 from [redacted]
prbit@localhost:~$
```

Finálna otázka

- Aké je najvhodnejšie poradie nastavení (úloh) z hľadiska bezpečnosti?

Po zriadení VM je najlepšie čo najskôr zakázať prihlasovanie na roota s tým, že si vygenerujeme kľúč na pripájanie a zakážeme aj pripájanie pomocou hesla. To by malo zaručiť, že iba my ako jediná osoba (resp. každá osoba s daným kľúčom) sa vie pripojiť na server. Následne by sme mali zistiť informácie o sieti a podľa toho nakonfigurovať firewall.